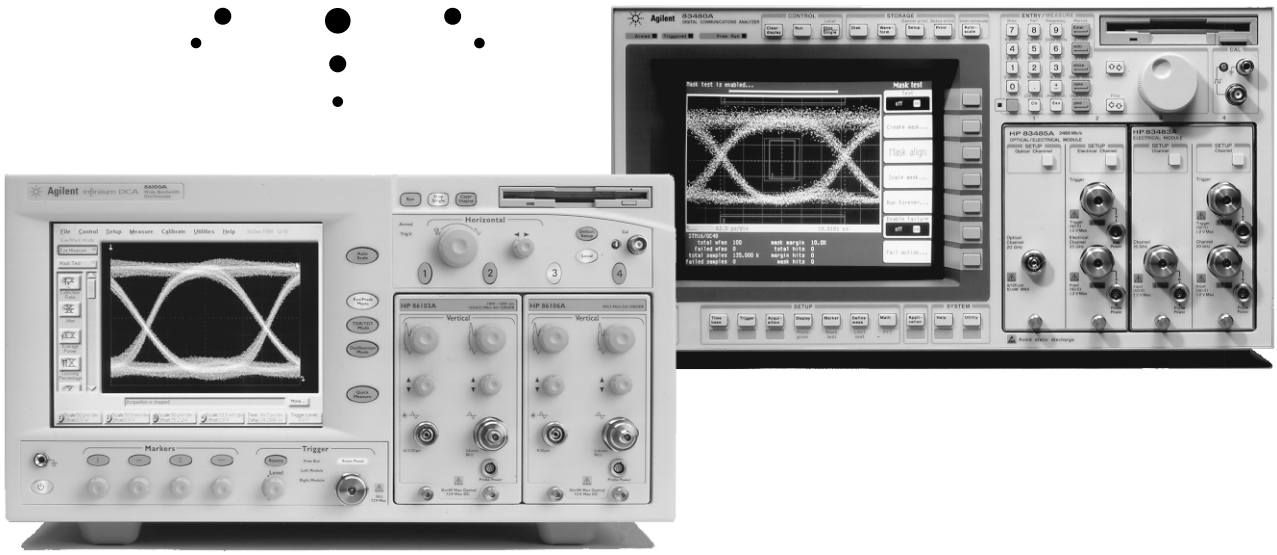
Converting ATE Control from the Agilent 83480A to 86100A

Product Note 86100-1



The Agilent Technologies 86100A Digital Communications Analyzer (DCA) will not provide perfect compatibility with the 83480A. In particular, it isn't possible to directly port all 83480A programming mnemonics to the 86100A.

Instead, the aim is to make it easy to recompile ATE programs to run on the 86100A. Agilent Technologies is unable to provide perfect compatibility for several reasons:

1. Technical difficulty in the case of architecture changes
2. Low return / low utilization of some other features
3. Attempts to simplify ATE Control of the DCA

Most programs should port easily, but some programmer attention will be required.

The Appendix includes some code examples written in BASIC that will assist you in writing new code for the 86100A, or to make your ATE programs compatible with both platforms.

It is assumed that the reader understands the elements of syntax of the 83480A.

Recommended Literature	**Agilent Part Number**
83480A Programmer's Guide	83480-90023
83480A User's Manual	83480-90027
86100A Programmer's Guide	86100-90003
86100A On-line Documentation	

Agilent Technologies
Innovating the HP Way

Table of Contents

Syntax Changes

:CHANnelN 4
Channel Autoscale 4
Channel Units 4
:DISPlay 4
Controlling the Color Graded Display 4
Displaying Labels 4
Displaying Inverse Labels 4
Clearing the Text Labels 4
:HISTogram 4
Setting the Histogram Scale 4
:MARKer 4
Cursor? Query 4
Marker Mode 4
Marker Units 5
:MTESt - Mask Test 5
FILL Display Mask Alignment 5
Counting Failures Within Polygons 5
Counting With Margins 5
Starting the Mask Test 5
Loading Masks 5
:MEASure 6
Extinction Ratio Calibration 6
:TRIGger 6
Trigger Source and Sweep 6
Trigger Level 6
:TRIGgerN 6

New or Enhanced 86100A Commands

ROOT 7
:ACQuire 7
:CALibrate 7
:DISK 7
:DISPlay 7
:FUNCtion 7
:HARDcopy 7
:HISTogram 7
:MARKer 7
:MEASure 7
:MTESt 7
:TRIGger 7

Commands Not Supported by the 86100A

Root Level Commands 8
:CALibrate 8
:CHANnelN 8
:DISK 8
:DISPlay 8
:FFT 8
:HARDcopy 9
:HISTogram 9
:LTESt - Limit Test 9
:MARKer 9
:MTESt - Mask Test 9
:MEASure 10
:PMEMory 10
:SERVice 10

:SYSTem ... 10
:TIMebase ... 10
:TRIGger ... 10
:TRIGgerN ... 10
:WAVeform ... 10
:WMEMory - Waveform Memory ... 10

Appendix

LATE BREAKING NEWS ... 11
Sample Code ... 11
Which Platform? ... 11
Channel Autoscale ... 12
Setting the Trigger Level ... 12
Setting the Trigger Source ... 12
Adjusting the Trigger Bandwidth ... 13
A Customized VUPper Measurement ... 14
An Efficient Histogram Mean Measurement ... 14
Tips and Tidbits ... 14
Quick Reference ... 15
Common ... 15
Root ... 15
:ACQuire ... 16
:CALibrate ... 16
:CHANnel<N> ... 16
:DISK ... 17
:DISPlay ... 17
:FUNCtion<N> ... 17
:HARDcopy ... 18
:HISTogram ... 18
:LTESt ... 18
:MARKer ... 19
:MEASure ... 19
:MTESt ... 20
:SYSTem ... 20
:TIMebase ... 20
:TRIGger ... 21
:WAVeform ... 21
:WMEMory(1:4) ... 21

Syntax Changes

These are commands that have changed from the 83480A to the 86100A. No functionality is lost.

:CHANneIN

Channel Autoscale
83480A :CHANneIN:AUToscale
86100A :AUToscale
The 86100A :AUToscale command optimizes the trigger level, horizontal settings, and vertical amplitude of the active channel(s). If the trigger level and/or horizontal settings are required to stay fixed, they can first be queried, and then reset after the :AUToscale command is executed. Refer to the channel_autoscale_86100a example in the Appendix.

Channel Units
Attenuation is no longer supported.

:DISPlay

Controlling the Color Graded Display
83480A :DISPlay:CGRade ON
86100A :DISPlay:PERSistence CGRade or :SYSTem:MODE EYE
The :SYSTem:MODE EYE command places the 86100A into a special mode of operation specifically suited for Eye Diagram measurements. The display persistence is adjusted for a new "Gray Scale" setting, similar to infinite persistence. If traditional 83480A color grade is desired in Eye Mode, use the following syntax:

86100A :SYSTem:MODE EYE;:DISPlay:PERSistence CGRade

Displaying Labels
83480A :DISPlay:COLumn <value>;ROW <value>;STRing "Label String"
86100A :DISPlay:LABel "Label String",<row>,<column>

Displaying Inverse Labels
83480A :DISPlay:INVerse;STRing "Label String"
86100A :DISPlay:LABel "Label String",<row>,<column>,<text_color>,<background>

Clearing the Text Labels
83480A :DISPlay:TEXT BLANk
86100A :DISPlay:LABel:DALL

:HISTogram

Setting the Histogram Scale
The 83480A specified the size of the histogram in scale per division. So, for a histogram that displayed 100% in four divisions, the scale would be 25%/div. The 86100A specifies display size of the histogram in divisions.

83480A :HIST:SCALe:SCALe 25%
86100A :HIST:SCALe:SIZe 4

:MARKer

Cursor? Query
The cursor query for the 83480A returns two values, time and amplitude. For the 86100A use the marker position queries.

83480A :MARKer:CURSor? STARt;CURSor? STOP;CURSor? DELTA
86100A :MARKer:X1P?;Y1P?;X2P?;Y2P?;XDELta?;YDELta?

Marker Mode
The 83480A :MARKer:MODE commands have been consolidated with the new 86100A :MARKer:STATe commands for the 86100A. Some of the 83480A :MARKer:MODE options are not required for remote operation and are not implemented in the 86100A, such as :MARKer:MODE MEASurement. The 83480A only allowed control of both sets of markers as a group. The 86100A provides control of the markers separately.

83480A :MARKer:MODE {HISTogram | MANual | MEASurement | OFF | WAVeform}
86100A :MARKer:STATe X1Y1 (for marker set 1)
:MARKer:STATe X2Y2 (for marker set 2)

Marker Units
Markers are associated with a particular source using the :MARK:X1Y1source or :MARK:X2Y2source commands. Querying the units of the marker's source can identify its units.

83480A :MARKer:X1Y1source CHANnelN;YUNits?
86100A :WAVeform:SOURce CHANnelN;YUNits?

:MTESt - Mask Test

The majority of changes for the Mask subsystem are involved with the actual mask file format. Masks created for the 83480A are not compatible with the 86100A. Also, the :MASK:DEFine and :MASK:POLYgon subsystems are not currently supported.

FILL Display Mask Alignment
The generic :AUToscale and FILL display mode of aligning masks was too slow. The 86100A combines this functionality in the new generic :AUToscale mode.

83480A :MTESt:AMODe FILL
86100A :AUT;:MTESt:ALIGn

Counting Failures Within Polygons
The new syntax for counting failures in the 86100A is the new REGION parameter. Using the POLYGONS parameter is maintained for compatibility only.

83480A :MTESt:COUNt:FAILures? POLYgon<n>
86100A :MTESt:COUNt:FAILures? REGion<n>

Counting With Margins
The 86100A definition of the parameters to the :MTESt:COUNt:HITS? Command is slightly different from the 83480A.

83480A and 86100A :MTESt:COUNt:HITS? {MARGin | MASK | TOTal}

The 83480A and 86100A return the same information for a POSITIVE (+) margin. The MARGin parameter returns the number of samples in the space BETWEEN the outer margin and the standard mask. The MASK parameter returns the number of samples within the standard mask region. The TOTal parameter returns the sum of TOTal=MARGin+MASK, that is, all hits within the margin space and standard mask region.

For NEGATIVE (-) margins, the answers are different. For the MARGin parameter, the 83480A returns the number of samples in the space BETWEEN the inner (-) margin and the outer standard mask, and the TOTal=MASK. The 86100A returns the number of samples only WITHIN the inner (-) margin space, and the TOTal=MARGin+MASK.

To summarize, for the MARGin parameter, the 83480A always returns the samples counted within the space between the standard mask and the perimeter of the margin for POSITIVE (+) or NEGATIVE (-) margins. The 86100A returns samples between the margin perimeter and the standard mask for POSITIVE (+) margins, and samples contained only within the margin perimeter for NEGATIVE (-) margins.

Starting the Mask Test
The new syntax for initiating the mask test in the 86100A is the new STARt mnemonic. Using the TEST mnemonic is maintained for compatibility only.

83480A :MTESt:TEST {{ON | 1} | {OFF | 0}}
86100A :MTESt:STARt
:MTESt:EXIT

Loading Masks
The 83480A maintained three methods of loading predefined masks. The 86100A consolidated all mask loading within internally stored mask files. The :DISK:LOAD syntax is maintained in the 86100A for compatibility, but the preferred syntax is the new :MTESt:LOAD command.

83480A	:MTESt:RECall <n>	To recall a mask from internal storage
	:MTESt:STANdard <standard>	To recall one of several pre-defined standard masks
	:DISK:LOAD "<filename>",[MASK]	To recall a mask from floppy storage
86100A	:MTESt:LOAD "filename"	

:MEASure

Most of the measurement subsystem is available with the 86100A. The few exceptions can be accomplished using the other commands built in to the 86100A, including the use of the histograms. Examples are included in the Appendix. The following measurements are not supported in the 86100A but can be made using other methods built in to the 86100A. Refer to the 83480A User's Guide for detailed information on the definition of these measurements.

:MEASure:FREQuency
:MEASure:NWIDth
:MEASure:PREShoot
:MEASure:PWIDth
:MEASure:STATistics
:MEASure:VUPper
:MEASure:VMIDdle
:MEASure:VLOWer

Extinction Ratio Calibration
The Extinction Ratio calibration function was contained in the :MEASure subsystem, but has been moved to the :CALibrate subsystem in the 86100A.

83480A :MEASure:CGRade:ERCalibrate
86100A :CALibrate:ERATio:STARt CHANnelN
:CALibrate:CONTinue or :CALibrate:CANCel

The status of the dark calibration can be checked with :CALibrate:ERATio:STATus <CHANnelN> and the dark level that was characterized during the calibration can be queried with :CALibrate:ERATio:DLEVel? <CHANnelN>.

:TRIGger

The trigger subsystem was changed slightly, to add support for the new front panel input, and to simplify the remaining vintage module programming.

Trigger Source and Sweep
The 83480A had two functional sources the trigger on channel 2 of the left module, or channel 4 of the right module. For the sweep type, :TRIGgered or FREerun. The 86100A-default trigger is the new front panel connector, and identifies "Freerun" as a source, rather than a trigger mode. The 86100A also provides support for 83480A modules, so it can identify those modules as trigger sources.

83480A :TRIGger:SOURce TRIGgerN
:TRIGger:SWEep {TRIGgered | FREerun }
86100A :TRIGger:SOURce {FPANel | LMODule | RMODule | FRUN}

Trigger Level
The 83480A specified different trigger levels for each of the trigger sources. The trigger source was identified as TRIGger2 for the trigger input on the left module or TRIGger4 for the right module.

The Agilent 86100A specifies only one trigger level, no matter what source.

83480A :TRIGger:LEVel TRIGgerN,<level>
86100A :TRIGger:LEVel <level>

:TRIGgerN

Similar to the source and sweep commands, the 86100A simplifies the bandwidth settings. The 83480A specified which trigger channel had a specific bandwidth, the 86100A specifies bandwidth for the entire trigger system. The 86100A also separates the :GATed function into a new subcommand.

83480A :TRIGgerN:BWLimit {ON | 1 | OFF | 0 | DIVided | GATE}
86100A :TRIGger:BWLimit {HIGH | LOW | DIVided}
:TRIGger:GATed {ON | 1 | OFF | 0}

New or Enhanced 86100A Commands

The following subsystem commands are new or have enhanced functionality. Please refer to the 86100A Programming Manual for detailed information.

ROOT

:AEEN - Replaces HEEN
:AEEN? - Replaces HEEN?
:ALER? - Replaces HEER?

:ACQuire

:ACQuire:RUNTil
:ACQuire:SSCReen
:ACQuire:SSCReen:AREA
:ACQuire:SSCReen:IMAG
:ACQuire:SWAVeform

:CALibrate

:CALibrate:CANCel
:CALibrate:CONTinue
:CALibrate:MODule
:CALibrate:ERATio
:CALibrate:SDONe?

:DISK

:DISK:PWD?
:DISK:CDIRectory
:DISK:MDIRectory

:DISPlay

:DISPlay:DATA?
:DISPlay:PERSistence
:DISPlay:LABel
:DISPlay:SSAVer

:FUNCtion

Two more function traces are available with the 86100A, for a total of four.

:HARDcopy

:HARDcopy:IMAGe
:HARDcopy:PRINters?
:HARDcopy:DPRinter

:HISTogram

:HISTogram:SCALe:SIZe

:MARKer

:MARKer:STATe

:MEASure

:MEASure:ANNotation

:MTESt

:MTESt:EXIT
:MTESt:LOAD “filename”
:MTESt:SCALe:YTRack
:MTESt:STARt

:TRIGger

:TRIGger:GATed
:TRIGger:BWLimit

Commands Not Supported by the 86100A

Root Level Commands

:AER? was underutilized.

:ERASe was used with pixel memory and was underutilized. Similar functionality is achieved with infinite persistence, gray scaled, or color graded displays. See :DISPlay:PERSistence

:MENU was underutilized.

:MERGe was used with pixel memory and was underutilized. Similar functionality is achieved with infinite persistence, gray scaled, or color graded displays. See :DISPlay:PERSistence

:STORe:PMEMory was underutilized. Similar functionality is achieved with infinite persistence, gray scaled, or color graded displays. See :DISPlay:PERSistence

:CALibrate

:CALibrate:FRAMe was underutilized

:CALibrate:PLUGin is changed to :CALibrate:MODule

:CHANnelN

:CHANnelN:AUToscale can be accomplished in code.

:CHANnelN:PROBe has moved to the :CALibrate subsystem. See :CALibrate:PROBe

:CHANnelN:SKEW has moved to the :CALibrate subsystem. See :CALibrate:SKEW

:CHANnelN:UNITs, originally included for external o/e devices, was underutilized.

:DISK

:DISK:FORMat was underutilized. Any DOS-compatible system can format floppy disks for use in the 86100A.

:DISPlay

:DISPlay:ASSign was underutilized. Similar functionality is available by using channel offsets and scales.

:DISPlay:CGR {on | off} has been moved to the :DISPlay:PERSistence subsystem.

:DISPlay:COLumn is consolidated under the :DISPlay:LABel subsystem.

:DISPlay:FORMat was underutilized. Similar functionality is available by using channel offsets and scales.

:DISPlay:INVerse is consolidated under the :DISPlay:LABel subsystem.

:DISPlay:LINE is consolidated under the :DISPlay:LABel subsystem.

:DISPlay:MASK was underutilized.

:DISPlay:ROW is consolidated under the :DISPlay:LABel subsystem.

:DISPlay:SOURce was used with pixel memory and was underutilized. Similar functionality is achieved with infinite persistence, gray scaled, or color graded displays. See :DISPlay:PERSistence.

:DISPlay:STRing is consolidated under the :DISPlay:LABel subsystem.

:DISPlay:TEXT is consolidated under the :DISPlay:LABel subsystem.

:FFT

:FFT was underutilized.

:HARDcopy

The entire :HARDcopy subsystem was changed to accommodate the MS-Windows® compatible printing system. All :HARDcopy outputs can be fully configured using the new 86100A commands working on top of the MS-Windows operating system. The following 83480A commands are not required:

:HARDcopy:ADDRess
:HARDcopy:BACKground
:HARDcopy:DESTination
:HARDcopy:DEVice
:HARDcopy:FFEed
:HARDcopy:FILename
:HARDcopy:LENGth
:HARDcopy:MEDia

:HISTogram

:HISTogram:RRATe not required. The 86100A optimizes the refresh rate internally.
:HISTogram:SCALe consolidated into :HISTogram:SIZE

:LTESt - Limit Test

:LTESt:SSCReen:DPRinter
:LTESt:SSUMmary:ADDRess;FORMat;MEDia;PFORmat;PORT;

:MARKer

:MARKer:CURSor?
:MARKer:MEASurement:READout - Use :MEASure:ANNotation
:MARKer:MODE
:MARKer:XUNits?
:MARKer:YUNits?

:MTESt - Mask Test

:MTESt:AMASk:Create - The AMASk subsystem is not currently supported.
:MTESt:AMASk:SOURce
:MTESt:AMASk:UNITs
:MTESt:AMASk:XDELta
:MTESt:AMASk:YDELta
:MTESt:AMODe - There is only one Mask Alignment mode - Align to waveform.
:MTESt:DEFine - Custom masks are not currently supported.
:MTESt:POLYgon:DEFine
:MTESt:RECall - Masks can be recalled from files on the internal hard disk.
:MTESt:SAVE - Saving masks is not currently supported.
:MTESt:STANdard - STANdard is no longer supporeted.

Windows is a U.S. registered trademark of Microsoft Corporation.

:MEASure

:MEASure:CGRade:ERCalibrate - Moved to the CALibrate subsystem.
:MEASure:CGRade:ERFactor - Moved to the CALibrate subsystem.
:MEASure:DUTYcycle - Underutilized, can be accomplished with TEDGe.
:MEASure:FFT - Underutilized.
:MEASure:FREQuency - Underutilized, use TEDge plus code.
:MEASure:NWIDth - Underutilized, use TEDge plus code.
:MEASure:PREShoot - Underutilized, use TEDge plus code.
:MEASure:PWIDth - Underutilized, use TEDge plus code.
:MEASure:STATistics - Statistics are always on for the 86100A.
:MEASure:VUPper - Underutilized, use TEDge plus code.
:MEASure:VMIDdle - Underutilized, use TEDge plus code.
:MEASure:VLOWer - Underutilized, use TEDge plus code.

:PMEMory

Pixel memory is not supported in the 86100A.

:SERVice

The 83480A Service Commands are not implemented in the 86100A.

:SYSTem

:KEY was underutilized.

:TIMebase

:TIMebase:VIEW was underutilized.
:TIMebase:WINDow was underutilized.

:TRIGger

:TRIGger:SWEep was consolidated in :TRIGger:SOURce.

:TRIGgerN

:TRIGgerN:BWLimit was consolidated in :TRIGger:BWLimit.

:WAVeform

:WAVeform:COUPling
:WAVeform:COMPlete
:WAVeform:VIEW

:WMEMory - Waveform Memory

All 83480A waveform memory commands are implemented in the 86100A.

Appendix

Late Breaking News

The following test limit commands are included in the 86100A.

:MTEE
:MTEE?

:MTER?

:MTESt:{RUNTil | RUMode} {{OFF | FORever} | FSAMples,<num_failed_samples>}
OFF is the preferred form. FORever is retained for backward compatibility.

:MTESt:{RUNTil | RUMode}?
RUNTil is the preferred form. RUMode is retained for backward compatibility.

:MTESt:SSCReen {OFF | DISK[,"filename"]}
:MTESt:SSCReen?

:MTESt:SSCReen:AREA {GRATicule | SCReen}
:MTESt:SSCReen:AREA?

:MTESt:SSCReen:IMAG {MONochrome | NORMal | INVert}
:MTESt:SSCReen:IMAG?

:MTESt:SWAVeform {CHANnel<N> | FUNCtion<N> | WMEMory<N>},{OFF | DISK[,"filename"[,{INTernal | TEXT[,{VERBose | YVALues}]}]]}
:MTESt:SWAVeform?

:LTEE
:LTEE?

:LTER?

:LTESt:{RUNTil | RUN | RUMode} {{OFF | FORever} | FAILures,<num_failures>}
OFF is the preferred form. FORever is retained for backward compatibility.

:LTESt:{RUNTil | RUN | RUMode}?
RUNTil is the preferred form. RUN and RUMode are retained for backward compatibility.

:LTESt:SSCReen {OFF | DISK[,"filename"]}
:LTESt:SSCReen?

:LTESt:SSCReen:AREA {GRATicule | SCReen}
:LTESt:SSCReen:AREA?

:LTESt:SSCReen:IMAG {MONochrome | NORMal | INVert}
:LTESt:SSCReen:IMAG?

:LTESt:SWAVeform {CHANnel<N> | FUNCtion<N> | WMEMory<N>},{OFF | DISK[,"filename"[,{INTernal | TEXT[,{VERBose | YVALues}]}]]}
:LTESt:SWAVeform?

Sample Code

Which Platform?
This example shows how to determine the instrument platform, so the appropriate commands can be used, using HP-BASIC®. Line numbers have been emitted for clarity.

```
DEF FNGet_model$( @Dca )        ! Pass in I/O to the instrument
  DIM Id$[255]
  !
  OUTPUT @Dca;"*IDN?"           ! Query for the Identity String
  ENTER @Dca;Id$                ! Read the Identity String
  !
```

HP BASIC is a U.S. registered trademark of Hewlett-Packard Company.

```
  IF POS(Id$,"83480A") THEN       ! See if the Identity String contains "83480A"
    RETURN "83480A"               ! Exit here and return "83480A" if found
  END IF
  !
  IF POS(Id$,"86100A") THEN       ! See if the Identity String contains "86100A"
    RETURN "86100A"               ! Exit here and return "86100A" if found
  END IF
  !
  DISP "ERROR: Did not find '83480A' or '86100A' in function FNGet_model$!"
  STOP                            ! Critical error - stop here
  !
FNEND
```

Channel Autoscale

This example saves the trigger and horizontal settings prior to performing :AUToscale so they can be re-set.

```
SUB channel_autoscale_86100a(@Dca)
  REAL Level,Delay,Scale
  !
  OUTPUT @Dca;":TRIGger:LEVel?"                          ! Ask for trigger level
  ENTER @Dca;Level
  OUTPUT @Dca;":TIMebase:DELay?;SCALe?"                  ! Ask for horizontal delay and scale
  ENTER @Dca;Delay,Scale
  OUTPUT @Dca;":AUToscale"                               ! Perform Autoscale
  OUTPUT @Dca;":TRIGger:LEVel",Level                     ! Re-set trigger level
  OUTPUT @Dca;":TIMebase:DELay",Delay;";SCALe",Scale     ! Re-set delay and scale
  !
SUBEND
```

Setting the Trigger Level

This subprogram selects different mnemonics to set the trigger level for either DCA platform.

```
SUB Set_trigger_level( REAL Level, @Dca ) ! Pass in Trigger Level and I/O Path
  DIM Trigger_source$[15]
  !
  IF FNGet_model$(@Dca)="83480A" THEN
    OUTPUT @Dca;":SYST:LONG OFF;HEAD OFF"
    OUTPUT @Dca;":TRIG:SOUR?" ! Get the active source, should be "TRIG2" or "TRIG4"
    ENTER @Dca;Trigger_source$
    OUTPUT @Dca;":TRIG:LEV ";Trigger_source$;",";Level
  ELSE ! Must be "86100A"
    OUTPUT @Dca;":TRIG:LEV ";Level ! Easy as pie
  END IF
SUBEND
```

Setting the Trigger Source

The following example shows how to implement a Trigger Setup routine that works for either the 86100A or 83480A. The function could use the new 86100A syntax to program either the 83480A or 86100A. No error checking is performed, but it would be simple to ensure the correct syntax is used for each model.

```
SUB Set_trigger_source( Trigger_source$, @Dca ) ! Pass in Trigger Source and I/O Path
  ! Use new 86100A mnemonics for Trigger_source$
  ! FPANel - Front Panel (86100A only)
  ! LMODule - Left Module (Used to be TRIGger2 for 83480A)
  ! RMODule - Right Module (Used to be TRIGger4 for 83480A)
  ! FRUN - Free Run (Used to be :TRIGger:SWEep FREerun for 83480A)
  !
  DIM Mnemonics$[255]
  !
  ! Trim and truncate Trigger_source$
```

```
  LET Trigger_source$=UPC$(TRIM$(Trigger_source$))
  LET Trigger_source$=Trigger_source$[1:4]
  IF FNGet_model$(@Dca)="83480A" THEN
    SELECT Trigger_source$
    CASE "RMOD"
      LET Mnemonics$=":TRIG:SOUR TRIG4;SWE TRIG"
    CASE "FRUN"
      LET Mnemonics$=":TRIG:SWE FRE"
    CASE ELSE ! Default to the Left Module
      LET Mnemonics$=":TRIG:SOUR TRIG2;SWE TRIG"
    END SELECT
  ELSE ! Must be "86100A"
    SELECT Trigger_source$
    CASE "LMOD"
      LET Mnemonics$=":TRIG:SOUR LMOD"
    CASE "RMOD"
      LET Mnemonics$=":TRIG:SOUR RMOD"
    CASE "FRUN"
      LET Mnemonics$=":TRIG:SOUR FRUN"
    CASE ELSE ! Default to the Front Panel
      LET Mnemonics$=":TRIG:SOUR FPAN"
    END SELECT
  END IF ! "83480A"
  OUTPUT @Dca;Mnemonics$ ! Send the code
SUBEND
```

Adjusting the Trigger Bandwidth

```
SUB Set_trigger_bandwidth(Bandwidth$,@Dca)
  !
  ! {HIGH | LOW | DIVIDED}
  !
  DIM Source$[15],Mnemonics$[255]
  LET Bandwidth$=UPC$(TRIM$(Bandwidth$))
  IF FNGet_model$(@Dca)="83480A" THEN
    OUTPUT @Dca;":SYST:LONG OFF;HEAD OFF"
    OUTPUT @Dca;":TRIG:SOUR?" ! Get the active source, should be "TRIG2" or "TRIG4"
    ENTER @Dca;Source$
    SELECT Bandwidth$[1;3]
    CASE "DIV"
      LET Mnemonics$=":"&Source$&":BWL DIV"
    CASE "LOW","ON","1"
      LET Mnemonics$=":"&Source$&":BWL ON"
    CASE ELSE ! Default - HIGH
      LET Mnemonics$=":"&Source$&":BWL OFF"
    END SELECT
  ELSE ! Must be an 86100A
    SELECT Bandwidth$[1:3]
    CASE "DIV"
      LET Mnemonics$=":TRIG:BWL DIV"
    CASE "LOW"
      LET Mnemonics$=":TRIG:BWL LOW"
    CASE ELSE ! Default - HIGH
      LET Mnemonics$=":TRIG:BWL HIGH"
    END SELECT
  END IF ! "83480A"
  OUTPUT @Dca;Mnemonics$ ! Send the code
SUBEND
```

A Customized VUPper Measurement

This example uses the built-in :MEASure:VTOP and :MEASure:VBASe commands to determine VUPper. VUPper is defined as the voltage value at the upper threshold of the waveform. For example, if the thresholds are set to 20% and 80%, then the VUPper value is the voltage at 80% of the maximum voltage on the rising or falling edge of the waveform.

```
DEF FNVUpper(@Dca)
  INTEGER Upper_threshold
  REAL Vtop,Vbase
  !
  ! First, set the thresholds.
  !
  LET Upper_threshold=0.8          ! 80% for upper threshold
  !
  ! Then, get VBASe (0%) and VTOP (100%)
  !
  OUTPUT @Dca;":MEASure:VTOP?;VBASe?"
  ENTER @Dca;Vtop,Vbase
  RETURN Vbase+Upper_threshold*(Vtop-Vbase)
FNEND
```

An Efficient Histogram Mean Measurement

```
write ":HEEN?;:OPEE?"          // Query and...
read heen, opee                // Store HEEN, OPEE
write ":HEEN 1;:OPEE 512"      // Make HER active, OPER mask = Histogram
write "*SRE?"                  // Query and...
read sre                       // Store SRE
write "*SRE 128"               // SRE mask = OPER
write ":HIST:RUNT?"            // Query and...
read runtil                    // Store RUNTil parms
write ":HIST:RUNT WAV,", n     // Upload new RUNTil parms
write "*CLS;:CDIS;:RUN;"       // Clear reg's, database, run
wait .05                       // Wait for gpib to catchup?
wait for spoll mask 0x40       // Wait for histogram completion
write ":MEAS:HIST:MEAN?"       // Query and...
read mean                      // Store mean
write ":HEEN ", heen           // Re-store HEEN
write ":OPEE ", opee           // Re-store OPEE
write "*SRE ", sre             // Re-store SRE
write ":HIST:RUNT ", runtil    // Re-store RUNTil parms
```

Tips and Tidbits

1. 83480A allowed multiple sources using color grade display. So does Sierra, but only one source is allowed in "eye mode".
2. 83480A returns 0,"No Error" in response to ":SYST:ERR? STR". 86100A returns 0,No Error (missing quotes).
3. Even though 86100A is in Eye/Grey Scale mode, remote measurements can still use "CGRade" as a source.
4. The Service Request Enable (SRE) Register is reset to +32 after power down/power up for both 83480A and 86100A. This register is an 8-bit register that can only be set to +191. That is, "*SRE 255;*SRE?" returns +191. The bit that is missing is #6, RQS/MSS, value 64.
5. The Event Status Enable (ESE) Register is reset to +255 after power down/power up for the 83480A, but it is reset to +0 by the 86100A.
6. The Trigger Event Register (TER) query returns 0 after power down/power up for the 83480A on the first query, but changes to 1 on subsequent queries if triggering. It is returns 1 by the 86100A if triggering.
7. The Trigger Event Enable Register (TEER) query returns -1 on the 83480A. It is not supported on the 86100A.
8. The User Event Enable (UEE) Register returns -1 after PD/PU for the 83480A, but the 86100A returns +0.
9. The Local Event Register (LER) returns +1 on the first query after PD/PU on the 83480A. The 86100A returns 0. Evidently the 83480A considered PD/PU a valid Remote-to-Local transition, the 86100A does not.

10. A 0.05 second wait is recommended prior to checking for SPOLL of a histogram RUNTil criteria, otherwise the SPOLL may return intermittently prior to the desired occurrence. The smallest wait that the SPOLL requires is approximately 0.025 seconds for the 86100A, and 0.05 seconds for the 83480A.
11. The 86100A returns "CALIBRATED" or "UNCALIBRATED" in response to a :CALibrate:MODule:STATus? query. The 83480A returns "NORMAL" or "UNKNOWN" in response to its :CALibrate:PLUGin:STATus? query.

Quick Reference

This quick reference is a subset of the most common DCA commands used. For the complete listing refer to the programming manual.

Common
*CLS
*ESE
*ESR?
*IDN?
*LRN
*OPC
*RCL
*RST
*SAV
*SRE
*STB?
*TRG
*TST?
*WAI

Root
:AUToscale
:BLANk {CHANnel<N> | FUNCtion<N> | WMEMory<N> | HISTogram}
:CDISplay
:COMMents {PLUGin<N> | LMODule | RMODule},<comments_text>
:COMMents? {PLUGin<N> | LMODule | RMODule}
:DIGitize {CHANnel<N> | FUNCtion<N>} [x10] What's this?
! DIGitize CHANnel<N> Turns off (BLANKs) all traces. :DIGitize does not.
:LTEE
:LTEE?
:LTER?
:MODel? {FRAMe | {LMODule | RMODule | PLUGin(1:4) | OS}
:MTEEnable <mask>
:MTEEnable?
:MTER?
:OPEEnable <mask>
:OPEEnable?
:OPERegister?
:PRINt
:RECALL:SETup <N>
:RUN
:SERial {FRAMe | PLUGin(1:4) | OS},<serial_number_string>
:SERial? {FRAMe | PLUGin(1:4) | OS}
:SINGle
:STOP
:STORe:SETup <N>
:STORe:WAVeform {CHANnel<N> | FUNCtion<N> | WMEMory<N>},WMEMory<N>
:TEER <mask> [undocumented]
:TEER? [undocumented]
:TERegister?

:UEEnable <mask>
:UEEnable?
:VIEW {CHANnel<N> | FUNCtion<N> | WMEMory<N> | HISTogram}

:ACQuire
:ACQuire:AVERage {{ON|1} | {OFF|0}}
:ACQuire:AVERage?
:ACQuire:COUNt <value>
:ACQuire:COUNt?
:ACQuire:POINts {AUTO | <points_value>}
:ACQuire:POINts?

:CALibrate
:CALibrate:OUTPut <value>
:CALibrate:OUTPut?
:CALirate:SKEW {CHANnel<N>},<value>
:CALirate:SKEW? {CHANnel<N>}
:CALibrate:CANCel
:CALibrate:CONTinue
:CALibrate:STATus?
:CALibrate:ERATio:STARt {CHANnel<N>}
:CALibrate:ERATio:STATus? {CHANnel<N>}
:CALibrate:ERATio:DLEVel? {CHANnel<N>}
:CALibrate:MODule:{ACCuracy | STATus}? {LMODule | RMODule | SLOT(1:4)}
:CALibrate:MODule:VERTical {LMODule | RMODule | SLOT(1:4)}
:CALibrate:MODule:CONTinue
:CALibrate:MODule:CANCel
:CALibrate:MODule:DONE?
:CALibrate:MODule:TIME? {LMODule | RMODule | SLOT(1:4)}
:CALibrate:{MODule | PLUGin}:MEMory? {LMODule | RMODule | SLOT(1:4)}
:CALibrate:{MODule | PLUGin}:OPOWer <value>
:CALibrate:{MODule | PLUGin}:OPOWer?
:CALibrate:MODule:OWAVelength <value>
:CALibrate:MODule:OWAVelength?
:CALibrate:MODule:OCONversion? {LMODule | RMODule | SLOT(1:4)}, {WAVelength1 | WAVelength2 | USER | W1310 | W1550}
:CALirate:SAMPlers {DISable | ENABle}
:CALirate:SAMPlers?

:CHANnel<N>
:CHANnel<N>:BANDwidth {HIGH | LOW}
:CHANnel<N>:BANDwidth?
:CHANnel<N>:DISPlay {ON | 1 | OFF | 0}
:CHANnel<N>:DISPlay?
:CHANnel<N>:FDEScription?
:CHANnel<N>:FILTer {ON | 1 | OFF | 0}
:CHANnel<N>:FILTer?
:CHANnel<N>:FSELect {FILTer1 | FILTer2}
:CHANnel<N>:FSELect?
:CHANnel<N>:OFFset <offset_value>
:CHANnel:OFFset?
:CHANnel<N>:PROBE <attenuation_factor>[,RATio | DECibel}] ! Undefined Header
:CHANnel<N>:RANGe <range_value>
:CHANnel<N>:RANGe?
:CHANnel<N>:SCALe <scale_value>
:CHANnel<N>:SCALe?
:CHANnel<N>:WAVelength {W1310 | W1550 | USER | WAVelength1 | WAVelength2}
:CHANnel<N>:WAVelength? ! Returns Undefined Header on non-optical channels

:DISK
:DISK:DELete “<filename>”
:DISK:DIRectory? [“<directory>” | {CGRade | ROOT | SETups | SIMages | WAVeforms }]
:DISK:PWD?
:DISK:CDIRectory “<directory>” | {CGRade | ROOT | SETups | SIMages | WAVeforms}
:DISK:MDIRectory “<directory>”

:DISPlay
:DISPlay:CGRade:LEVels?
:DISPlay:DATA? {BMP | PCX | EPS | PS | GIF},{SCReen | GRATicule},{ON | OFF}
:DISPlay:DCOLor
:DISPlay:GRATicule {GRID | FRAMe}
:DISPlay:GRATicule?
:DISPlay:GRATicule:INTensity <intensity>
:DISPlay:GRATicule:INTensity?
:DISPlay:PERSistence {CGRade | GSCale | MINimum | INFinite | <value>}
:DISPlay:PERSistence?
:DISPlay:SCOLor <color_name>, <hue>, <saturation>, <luninosity>
:DISPlay:SCOLor? <color_name>
:DISPlay:LABel “label_string”[,<row>[,<column>[,<text_color>[,<background>]]]]
:DISPlay:LABel:DALL
:DISPlay:SSAVer {DISabled | ENABled}
:DISPlay:SSAVer?
:DISPlay:SSAVer:AAFTer <time>
:DISPlay:SSAVer:AAFTer?

:FUNCtion<N>
:FUNCtion<N>?
:FUNCtion<N>:ADD <operand>,<operand>
:FUNCtion<N>:DIVide <operand>,<operand>
:FUNCtion<N>:MULTiply <operand>,<operand>
:FUNCtion<N>:SUBTract <operand>,<operand>
:FUNCtion<N>:VERSus <operand>,<operand>
:FUNCtion<N>:DIFF(erentiate)
:FUNCtion<N>:INTegrate
:FUNCtion<N>:INVert
:FUNCtion<N>:MAGNify
:FUNCtion<N>:MAXimum
:FUNCtion<N>:MINimum
:FUNCtion<N>:ONLY
:FUNCtion<N>:BWLimit
:FUNCtion<N>:{DISPlay} {ON | 1 | OFF | 0}
:FUNCtion<N>:{DISPlay}?
:FUNCtion<N>:HORizontal {TSOurce | AUTO | MANUAL}
:FUNCtion<N>:HORizontal?
:FUNCtion<N>:HORizontal:{POSition | OFFSet} <value> [OFFSet is undocumented]
:FUNCtion<N>:HORizontal:{POSition | OFFSet}? [OFFSet is undocumented]
:FUNCtion<N>:HORizontal:{RANGe | SCALe} <value> [SCALe is undocumented]
:FUNCtion<N>:HORizontal:{RANGe | SCALe}?
:FUNCtion<N>:OCHange {ON | 1 | OFF | 0} [undocumented]
:FUNCtion<N>:OFFSet <value>
:FUNCtion<N>:OFFSet?
:FUNCtion<N>:RANGe <value>
:FUNCtion<N>:RANGe?
:FUNCtion<N>:VERTical {TSOurce | MANual | AUTO}
:FUNCtion<N>:VERTical?
:FUNCtion<N>:VERTical:{POSition | OFFSet} <value> [POSition is undocumented]

:FUNCtion<N>:VERTical:{POSition | OFFSet}?
:FUNCtion<N>:VERTical:{RANGe} <value>
:FUNCtion<N>:VERTical:{RANGe}?

:HARDcopy
:HARDcopy:AREA {GRATicule | SCReen}
:HARDcopy:AREA?
:HARDcopy:IMAGe {NORMal | INVert | MONochrome}
:HARDcopy:IMAGe?
:HARDcopy:FACTors {ON | 1 | OFF | 0}
:HARDcopy:FACTors?
:HARDcopy:PRINters?
:HARDcopy:{DPRinter}

:HISTogram
:HISTogram:AXIS {VERTical | HORizontal}
:HISTogram:AXIS?
:HISTogram:MODE {OFF | WAVeform}
:HISTogram:MODE?
:HISTogram:RUNTil {FORever | WAVeform, <n_waveform> | SAMples, <n_samples> |}
:HISTogram:RUNTil?
:HISTogram:SCALe:SIZE <divisions>[,{HORizontal | VERTical}]
:HISTogram:SCALe:SIZE? [{HORizontal | VERTical}]
:HISTogram:WINDow:DEFault
:HISTogram:WINDow:SOURce {CHANnel<N> | FUNCtion<N>}
:HISTogram:WINDow:SOURce?
:HISTogram:WINDow:X1Position <position>
:HISTogram:WINDow:X1Position?
:HISTogram:WINDow:X2Position <position>
:HISTogram:WINDow:X2Position?
:HISTogram:WINDow:Y1Position <position>
:HISTogram:WINDow:Y1Position?
:HISTogram:WINDow:Y2Position <position>
:HISTogram:WINDow:Y2Position?

:LTESt
:LTESt:FAIL {INSide | OUTSide | ALWays | NEVer}
:LTESt:FAIL?
:LTESt:LLimit <lower_value>
:LTESt:LLimit?
:LTESt:MNFound {FAIL | PASS | IGNore}
:LTESt:MNFound?
:LTESt:{RUN | RUMode} {FORever | FAILures, <total_failures> | WAVeforms, <maximum_waveforms>}
:LTESt:{RUN | RUMode}?
:LTESt:SOURce {1 | 2 | 3 | 4} ! Numbers correspond to displayed measurements.
:LTESt:SOURce?
:LTESt:SSCReen {OFF | PRINter | DISK [,<filename>]}
:LTESt:SSCReen?
:LTESt:SSUMmary {OFF | DISK | PRINter}
:LTESt:SSUMmary?
:LTESt:SWAVeform <source>,<destination>
:LTESt:SWAVeform? <source>
:LTESt:TEST {ON | 1 | OFF | 0}
:LTESt:TEST?
:LTESt:ULIMit <upper_limit>
:LTESt:ULIMit?

:MARKer
:MARKer:STATe {X1Y1 | X2Y2}, {{OFF, {OFF | MANual}} | {MANual, {OFF | MANual | TRACk}}}
:MARKer:STATe? {X1Y1 | X2Y2}
:MARKer:{TDELta | XDELta}?
:MARKer:{TSTArt | X1Position} <X1_position>
:MARKer:{TSTArt | X1Position}?
:MARKer:{TSTOp | X2Position} <x2_position>
:MARKer:{TSTOp | X2Position}?
:MARKer:{VDELta | YDELta}?
:MARKer:{VSTArt | Y1Position} <y1_position>
:MARKer:{VSTArt | Y1Position}?
:MARKer:{VSTOp | Y2Position} <y2_position>
:MARKer:{VSTOp | Y2Position}?
:MARKer:X1Y1source {CHANnel<N> | FUNCtion<N> | WMEMory<N>}
:MARKer:X1Y1source?
:MARKEr:X2Y2source {CHANnel<N> | FUNCtion<N> | WMEMory<N>}
:MARKEr:X2Y2source?

:MEASure
:MEASure:APOWer {WATT | DECibel}, [{CHANnel(1:4)}]
:MEASure:CGRade:COMPlete <completion_hits>
:MEASure:CGRade:COMPlete?
:MEASure:CGRade:EHEight, CROSsing, ESN, QFACtor, OLEVel, ZLEVel
:MEASure:CGRade:EHEight, CROSsing, ESN, QFACtor, OLEVel, ZLEVel?
:MEASure:EHEight, CROSsing, ESN, OLEVel, ZLEVel
:MEASure:EHEight, CROSsing, ESN, OLEVel, ZLEVel?
:MEASure:CGRade:JITTer {PP | RMS}
:MEASure:CGRade:JITTer? {PP | RMS}
:MEASure:CGRade:ERATio {RATio | DECibel | PERCent}
:MEASure:CGRade:ERATio? {RATio | DECibel | PERCent}
:MEASure:CGRade:DCDistortion {TIME | PERCent}
:MEASure:CGRade:DCDistortion? {TIME | PERCent}
:MEASure:CGRade:EWIDth [{TIME | RATio}]
:MEASure:CGRade:EWIDth? [{TIME | RATio}]
:MEASure:CGRade:PEAK?
:MEASure:DEFine <meas_spec>
:MEASure:DEFine {DELTatime | EWINdow | THResholds | TOPBase}?
:MEASure:DELTatime [<source>[,<source>]]
:MEASure:DELTatime? [<source>[,<source>]]
:MEASure:{FALLtime | OVERshoot | PERiod | RISetime | VAMPlitude | VPP | EBITs| VBASe | VMAX | VMIN | VTOP | TMAX | TMIN } [<source>]
:MEASure:{FALLtime | OVERshoot | PERiod | RISetime | VAMPlitude | VPP | EBITs| VBASe | VMAX | VMIN | VTOP | TMAX | TMIN }? [<source>]
:MEASure:HISTogram:HITS?
:MEASure:HISTogram:SCALE}?
:MEASure:HISTogram:{PP | MEDian | MEAN | STDDev | M1S | M2S | M3S | HITS | PEAK}? [<source>]
:MEASure:RESults?
:MEASure:{SCRatch | CLEar} [CLEar is undocumented]
:MEASure:SENDvalid {ON | 1 | OFF | 1}
:MEASure:SENDvalid?
:MEASure:SOURce <source>[,<source>]
:MEASure:SOURce?
:MEASure:TEDGe {UPPer | MIDDle | LOWer | CROSsing},[<slope>]<occurrence>[,<source>]
:MEASure:TEDGe? {UPPer | MIDDle | LOWer | CROSsing},[<slope>]<occurrence>[,<source>]
:MEASure:TVOLt <voltage>,[<slope>]<occurence>[,<source>]
:MEASure:TVOLt? <voltage>,[<slope>]<occurence>[,<source>]
:MEASure:VAVerage {CYCle | DISPlay}[,{CHANnel<N> | FUNCtion<N> | WMEMory<N> | CGRade}]
:MEASure:VAVerage? {CYCle | DISPlay}[,{CHANnel<N> | FUNCtion<N> | WMEMory<N> | CGRade}]

:MEASure:VRMS {CYCle | DISPlay},{AC | DC}[,{CHANnel<N> | FUNCtion<N> | WMEMory<N> | CGRade}]
:MEASure:VRMS? {CYCle | DISPlay},{AC | DC}[,{CHANnel<N> | FUNCtion<N> | WMEMory<N> | CGRade}]
:MEASure:VTIMe <time>[,{CHANnel<N> | FUNCtion<N> | WMEMory<N>}]
:MEASure:VTIMe? <time>[,{CHANnel<N> | FUNCtion<N> | WMEMory<N>}]
:MEASure:ALL? {CHANnel<N> | FUNCtion<N> | HISTogram | WMEMory<N> | CGRade} [undocumented]

:MTESt
:MTESt:ALIGn
:MTESt:COUNt:FAILures? {POLYgon | REGIon}<n>
:MTESt:COUNt:FWAVeforms?
:MTESt:COUNt:FSAMples?
:MTESt:COUNt:HITS? {MARGin | MASK | TOTal}
:MTESt:COUNt:SAMPles?
:MTEst:COUNt:WAVeforms?
:MTESt:DELete
:MTESt:FILename?
:MTESt:LOAD “filename”
:MTESt:MASK:DELete
:MTESt:MMARgin:PERCent <margin_percent>
:MTESt:MMARgin:PERCent?
:MTESt:MMARgin:STATe {{ON | 1} | {OFF | 0}}
:MTESt:MMARgin:STATe?
:MTESt:SCALe:DEFault
:MTESt:SCALe:SOURce {CHANnel<n> | FUNCtion<n>}
:MTESt:SCALe:SOURce?
:MTESt:SCALe:X1 <x1_value>
:MTESt:SCALe:X1?
:MTESt:SCALe:XDELta <xdelta_value>
:MTESt:SCALe:XDELta?
:MTESt:SCALe:Y1 <y1_value>
:MTESt:SCALe:Y1?
:MTESt:SCALe:Y2 <y2_value>
:MTESt:SCALe:Y2?
:MTESt:STARt
:MTESt:TEST {{ON | 1} | {OFF | 0}
:MTESt:TITle?

:SYSTem
:SYSTem:DATE <day>,<month>,<year>
:SYSTem:DATE? ! 86100 returns <dd mmm yyyy>
:SYSTem:DSP <string>
:SYSTem:DSP? ! 86100 returns last string sent with :DSP
:SYSTem:ERRor? [{NUMBer | STRing}]
:SYSTem:HEADer {{ON | 1} | {OFF | 0}}
:SYSTem:LONGform {{ON | 1} | {OFF | 0}}
:SYSTem:MODE {EYE | OSCilloscope}
:SYSTem:MODE?
:SYSTem:SETup <binary block data>
:SYSTem:SETup?
:SYSTem:TIME <hour>,<minute>,<second>
:SYSTem:TIME?

:TIMebase
:TIMebase:BRATe <bit_rate>
:TIMebase:BRATe?
:TIMebase:{POSition | DELay} <delay_value> ! New minimum = 24 ns
:TIMebase:{POSition | DELay}? ! DEL is preferred
:TIMebase:RANGe <full_scale_range_value>
:TIMebase:RANGe? ! Always returns time

:TIMebase:REFerence {LEFT | CENTer}
:TIMebase:REFerence?
:TIMebase:SCALe <units_per_div | bits_per_display>
:TIMebase:SCALe?
:TIMebase:UNITs {TIME | BITS }
:TIMebase:UNITs?

:TRIGger
:TRIGger:HYSTeresis {NORMal | HSENsitivity}
:TRIGger:HYSTeresis?
:TRIGger:LEVEL <level>
:TRIGger:LEVEL?
:TRIGger:SLOPe {POSitive | NEGative}
:TRIGger:SLOPe?
:TRIGger:SOURce {FPANel | FRUN | LMODule | RMODule}
:TRIGger:SOURce?
:TRIGger:GATed {ON | 1 | OFF | 1}
:TRIGger:GATed?
:TRIGger:BWLimit {HIGH | LOW | DIVided}
:TRIGger:BWLimit?

:WAVeform
:WAVeform:BANDpass?
:WAVeform:BYTeorder {MSBFirst | LSBFirst}
:WAVeform:BYTeorder?
:WAVeform:COUNt?
:WAVeform:DATA?
:WAVeform:FORMat {ASCii | BYTE | LONG | WORD}
:WAVeform:FORMat?
:WAVeform:POINts?
:WAVeform:PREamble?
:WAVeform:SOURce {CHANnel<N> | FUNCtion<N> | WMEMory<N> | HISTogram | {DATabase | CGRade}}
:WAVeform:SOURce? ! If set to DAT or CGR returns “COL” or “COLORGRADE-GRAYSCALE”
:WAVeform:TYPE?
:WAVeform:XDISplay?
:WAVeform:XINCrement?
:WAVeform:XORigin?
:WAVeform:XRANge?
:WAVeform:XREFerence?
:WAVeform:XUNits?
:WAVeforma;YDISplay?
:WAVeform:YINCrement?
:WAVeform:YORigin?
:WAVeform:YRANge?
:WAVeforma;YREFerence?
:WAVeforma;YUNits?

:WMEMory(1:4)
:WMEMory<N>:DISPlay {{ON | 1} | {OFF | 0}}
:WMEMory<N>:DISPlay?
:WMEMory<N>:SAVE {CHANnel<N> | WMEMory<N> | FUNCtion<N>}
:WMEMory<N>:XOFFset <offset_value>
:WMEMory<N>:XOFFset?
:WMEMory<N>:XRANge <range_value>
:WMEMory<N>:XRANge?
:WMEMory<N>:YOFFset <offset_value>
:WMEMory<N>:YOFFset?
:WAVeforma;YRANge?

For more information about Agilent Technologies test and measurement products, applications, services, and for a current sales office listing, visit our web site,

www.agilent.com/comms/lightwave

You can also contact one of the following centers and ask for a test and measurement sales representative.

United States:
Agilent Technologies
Test and Measurement Call Center
P.O. Box 4026
Englewood, CO 80155-4026
(tel) 1 800 452 4844

Canada:
Agilent Technologies Canada Inc.
5150 Spectrum Way
Mississauga, Ontario
L4W 5G1
(tel) 1 877 894 4414

Europe:
Agilent Technologies
Test & Measurement
European Marketing Organisation
P.O. Box 999
1180 AZ Amstelveen
The Netherlands
(tel) (31 20) 547 9999

Japan:
Agilent Technologies Japan Ltd.
Measurement Assistance Center
9-1, Takakura-Cho, Hachioji-Shi,
Tokyo 192-8510, Japan
(tel) (81) 426 56 7832
(fax) (81) 426 56 7840

Latin America:
Agilent Technologies
Latin American Region Headquarters
5200 Blue Lagoon Drive, Suite #950
Miami, Florida 33126, U.S.A.
(tel) (305) 267 4245
(fax) (305) 267 4286

Australia/New Zealand:
Agilent Technologies Australia Pty Ltd
347 Burwood Highway
Forest Hill, Victoria 3131, Australia
(tel) 1-800 629 485 (Australia)
(fax) (61 3) 9272 0749
(tel) 0 800 738 378 (New Zealand)
(fax) (64 4) 802 6881

Asia Pacific:
Agilent Technologies
24/F, Cityplaza One, 1111 King's Road,
Taikoo Shing, Hong Kong
(tel) (852) 3197 7777
(fax) (852) 2506 9284

Technical data subject to change
Copyright © 2000
Agilent Technologies

Printed in U.S.A. 2/00
5968-9664E

Agilent Technologies
Innovating the HP Way